

PATENT  
RECEIVED  
CENTRAL FAX CENTER

JUL 28 2005

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:  
Barker et al.

Serial No.: 10/046,940

Filed: January 14, 2002

Title: System and Method for Mapping  
Management Objects to Console  
Neutral User Interface

§ Group Art Unit: 2179

§

§ Examiner: Huynh, Ba

§

§ Attorney Docket No. RSW920010049US1

§

§

§ IBM Corporation

§ Intellectual Property Law Dept. T81/503

§ 3039 Cornwallis Road

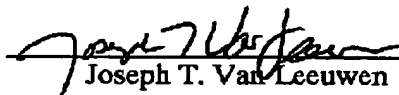
§ Research Triangle Park, NC 27709

§

Certificate of Facsimile Under C.F.R. §1.8(a)

I hereby certify that this correspondence was sent via facsimile to the United States Patent Office to telephone number (703) 872-9306 on June 28, 2005.

By:

  
Joseph T. Van Leeuwen7/28/05  
DateAPPELLANTS' BRIEFMail Stop Appeal Brief-Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Sir:

**INTRODUCTORY COMMENTS**

This brief is filed in support of the previously filed Notice of Appeal, filed in this case on April 28, 2005, which appealed from the decision of the Examiner dated January 28, 2005, finally rejecting claims 1-25. Please charge the required fee to IBM Corporation Deposit Account No. 09-0461.

A one month extension of time is believed to be necessary, payment for which is enclosed. If, however, a further extension of time is required, the extension is requested, and the undersigned hereby authorizes the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0461.

Docket No. RSW920010049US1

Page 1 of 24  
Barker et al. - 10/046,940

Atty Ref. No. IBM-R109

**PATENT****REAL PARTY IN INTEREST**

The real party in interest in this appeal is International Business Machines Corporation, which is the assignee of the entire right, title, and interest in the above-identified patent application.

**RELATED APPEALS AND INTERFERENCES**

With respect to other prior or pending appeals, interferences, or judicial proceedings that are related to, will directly affect, be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such prior or pending appeals, interferences, or judicial proceeding known to Appellants, Appellants' legal representative, or assignee.

**STATUS OF CLAIMS***1. Total number of claims in application*

There are 25 claims pending. Seven claims are independent claims (1, 8, 15, and 22-25), and the remaining claims are dependent claims.

*2. Status of all claims in application*

- Claims canceled: None
- Claims withdrawn from consideration but not canceled: None
- Claims pending: 1-25.
- Claims allowed: None
- Claims rejected: 1-25.

*3. Claims on appeal*

The claims on appeal are: 1-25.

**STATUS OF AMENDMENTS**

All amendments have been entered in this case. No amendments have been made to the claims after the Final Office Action.

Docket No. RSW920010049US1

Page 2 of 24  
Barker et al. - 10/046,940

Atty Ref. No. IBM-R109

## PATENT

**SUMMARY OF CLAIMED SUBJECT MATTER**

Appellants provide a concise summary of the claimed subject matter as follows. Claims 1, 8, 15, and 22-25 are independent claims. Note that claims 1-7, 22, and 23 are method claims, claims 8-14 and 24 are information handling system claims, and claims 15-21 and 25 are computer program product claims. Independent claims 8 and 15 include one or more means plus function limitations that correspond to the method steps set forth in independent claim 1. An information handling system capable of implementing Appellants' invention, as claimed in independent claims 8 and 24 is shown in Figure 18, and described in Appellants' specification on page 45, line 10 through page 46, line 24. Support for independent computer program product claims 15 and 25 is described in Appellants' specification on page 46, line 25 through page 47, line 14. In addition, support for each of the method steps and means plus function limitations of the independent claims are discussed below. The specific citations to Appellants' Figures and Specification are meant to be exemplary in nature, and do not limit the scope of the claims. In particular, the citations below do not limit the scope of equivalents as provided under 35 U.S.C. § 112, sixth paragraph.

The claimed invention is a method, information handling system, and computer program product for generating display information from a management definition data file (see e.g., Figure 9, specification page 27, line 5 through page 29, line 5; Figure 15, specification page 38, line 15 through page 40, line 20; Figure 16, specification page 40, line 21 through page 43, line 17; and Figures 17a-17b, specification page 43, line 18 through page 45, line 9) by receiving an element request, (see, e.g., Figure 15 element 1505, specification page 38, line 15 through page 40, line 20), locating a display name (see, e.g., Figure 15 elements 1515-1575, specification page 38, line 15 through page 40, line 20); retrieving qualifier values and data definitions corresponding to the display name (see, e.g., Figure 15 elements 1530 and 1580, specification page 38, line 15 through page 40, line 20); creating data elements using the data definitions (see, e.g., Figure 15 element 1575, specification page 38, line 15 through page 40, line 20); and writing the names to a display panel (see, e.g., Figure 9, elements 940-960, specification page 27, line 5 through page 29; and Figure 15 elements 1585 and 1590, specification page 38, line 15 through page 40, line 20). Independent claims 22, 24 and 25 add the limitations of identifying menu tab names from qualifier names, creating menu tabs, and writing tab labels to the menu

## PATENT

tabs (see, e.g., Figure 9, element 940, specification page 27, line 5 through page 29) to the limitations discussed for claims 1, 8, and 15. Independent claims 23, 24, and 25 further add the limitations of retrieving text labels corresponding to data elements, and writing the text labels in a display panel in a position adjacent to each text label's corresponding data element (see, e.g., Figure 9, element 980, specification page 27, line 5 through page 29).

As required by 37 C.F.R. §41.37(c)(1)(v), Appellants provide support from the specification for the means plus function elements of each dependent claim argued separately below.

Dependent claim 17 adds the means plus function limitation of associating a data element with an external data source (see, e.g., Figure 9, element 970, specification page 27, line 5 through page 29) to the limitations of independent claim 15.. Dependent claim 18 adds means plus function limitations for identifying menu tab names from qualifier names, creating menu tabs, and writing tab labels to the menu tabs (see, e.g., Figure 9, element 940, specification page 27, line 5 through page 29) to the limitations of independent claim 15. Dependent claim 20 adds means plus function limitations for retrieving text labels corresponding to data elements, and writing the text labels in a display panel in a position adjacent to each text label's corresponding data element (see, e.g., Figure 9, element 980, specification page 27, line 5 through page 29) to the limitations of independent claim 15.

**GROUND OF REJECTION TO BE REVIEWED ON APPEAL**

- A. Claim rejections under 35 U.S.C. § 102(e) and 103 rejecting claims 1-25 as being unpatentable over U.S. Patent Publ. 2003/0095142 to Patrizio et al. (hereinafter "Patrizio").
- B. The Examiner's improper refusal to consider Applicants' declaration under 37 CFR 1.131 swearing behind the Patrizio reference.

## PATENT

## ARGUMENTS -

Claims 1- 25 Are *Not* Anticipated By Patrizio and are therefore *Allowable*

As an initial matter, Appellants note a defect in both the Office Action and the Final Office Action. Namely, each office action states that claims 1-13 and 25 are rejected under 102(e) as being anticipated by Patrizio, however each office action discusses how Patrizio allegedly anticipates each of Appellants' claims (claims 1-25). Therefore, Appellants' arguments are directed to the rejection of each claim rather than the stated rejection of claims 1-13 and 25.

As a second matter, Appellants note that Patrizio is not prior art to Appellants' claimed invention, as evidenced by Appellants' Declaration under Rule 1.131 that effectively swears behind the reference. Appellants' Declaration is discussed in the next argument section directed at the Examiner's refusal to consider Appellants' Declaration. Despite the fact that Patrizio is an improper reference because it is *not prior art*, the Patrizio reference also *does not anticipate* Appellants' claims as alleged in the Final Office Action.

Each of Appellants' independent claims 1, 8, and 15 are each directed to generating display information from management definition data and each include the limitations of:

- receiving an element request from a user;
- locating a display name corresponding to the element request in a management definition object;
- retrieving one or more qualifier values and one or more data definitions corresponding to the display name, wherein the retrieving includes reading the management definition object;
- creating one or more data elements using the data definitions; and
- writing the qualifier values and data elements to a display panel.

The Final Office Action contends that Patrizio teaches each of these limitations. A thorough review of Patrizio, however, reveals that Patrizio clearly does not teach each of these limitations and, therefore, Patrizio simply does not anticipate Appellants' claims.

## PATENT

The Final Office Action contends that Patrizio teaches Appellants' first limitation, "receiving an element request from a user," and cites Patrizio, paragraphs 35 and 37. Those paragraphs of Patrizio read as follows:

[0035] According to one embodiment of the invention, a class schema is identified which defines the visual components of the GUI that should be modifiable. The class schema and the corresponding class instances are defined in managed object format (MOF) files. MOF files follow a standard format that is well known to those skilled in the art. It will be apparent to one skilled in the art that as the Common Information Model (CIM) technology evolves, other formats might be used.

...

[0037] Another feature of the present invention is gathering layouts and organizational information into a managed object format. In order to structure the MOF, a UML (unified modeling language) class diagram is developed. This class diagram is an illustration showing how the MOF file works and what would be contained inside this MOF file. For example, for the cluster property sheet described above, there is a MOF file which contains all of the necessary information representing that General tab, the cluster Packages tab, the Nodes tab, and the Network tab. Inside of a file that is internal to ServiceGuard Manager there is a MOF file which contains a description telling the program how a cluster property sheet should be rendered.

The paragraphs cited by the Examiner describe various aspects of management object format (MOF) files. Nowhere do the cited sections describe receiving a request for a user, nor do the cited sections describe processing an "element request" as claimed by Appellants. In the cited paragraphs, Patrizio describes general attributes of a MOF file including sections regarding "the General tab, the cluster Packages tab, the Nodes tab, and the Network tab." However, the sections cited in the Final Office Action as teaching Appellants' limitation simply do not teach or suggest receiving any request from a user.

Appellants' next limitation, "locating a display name corresponding to the element request in a management definition object," is also not taught or suggested by Patrizio. The Final Office Action cites the same paragraphs from Patrizio in support of the contention that Patrizio teaches this limitation. A review of the cited paragraphs, however, reveals that Patrizio does not teach or suggest adding a "display tab," nor does Patrizio teach or suggest locating any "display names" from the MOF file. In fact, the cited paragraphs of Patrizio are completely void of the word "display."

## PATENT

Appellants' third limitation in each of these independent claims is for "retrieving one or more qualifier values and one or more data definitions *corresponding to the display name*, wherein the retrieving includes reading the management definition object." As shown above, Patrizio does not teach or suggest "locating a display name" from the MOF file. It therefore follows that Patrizio cannot possibly teach or suggest "retrieving ... qualifier values and ... data definitions" corresponding to such display names. A review of cited paragraphs 38 and 39 of Patrizio show that, indeed, Patrizio provides no such teaching:

[0038] If a change to the layout of a cluster property sheet is required, the modifications are added to the MOF file. For instance, for an additional tab, further descriptions describing another tab are added to the MOF file and the code in the program would not need to be modified. Thus, the instances of the classes are modified in the MOF file, but the schema maintains generality and need not be modified. In one embodiment, the application is programmed using JAVA.TM. (JAVA is a trademark of Sun Microsystems, Inc.). The JAVA.TM. code that exists would read that definition file and would automatically render a new tab. Traditionally, the way this is done is to hard-code it in source code. Thus, JAVA.TM. code to specifically render all of the components needed for the cluster property sheet would need to be written and reintegrated into the existing code. In the preferred embodiment, the desired changes are entered into a pre-processor which checks the syntax and then generates the modified MOF file.

[0039] Referring now to FIGS. 9A and 9B, there is shown a class schema, generally designated by the reference numeral 900a and 900B, for the property sheets, pursuant to the principles and teachings of the present invention. Referring specifically to FIG. 9A, class CMGuiPSheet 910 defines the general property sheet, and has a class identifier (mapClassId), a title string, a title property name string, a version and a default height and width, as illustrated in the FIG. 9A. In the exemplary embodiment, there are three objects having property sheets: cluster, node and package. Therefore, there will be three instances of the CMGuiPSheet class in the MOF file that holds the instances of the defined classes. If a new object is defined, the schema 900a requires no modification. The instance MOF file would be modified to add the new property sheet instance and associated class instantiations.

The cited paragraphs of Patrizio describes changes to a "cluster property sheet" (para. 38) and a "class schema ... for the property sheets." The cited paragraphs do not teach or suggest "retrieving one or more qualifier values and one or more data definitions" corresponding to a located display name. While most anything in a computer can be "displayed," including Patrizio's "cluster property sheets," displaying such sheets is simply not the same as (1) receiving an element request from a user, (2) locating a display name corresponding to the

## PATENT

element request, and (3) retrieving qualifier values and data definitions corresponding to the display name, as claimed by Appellants. While Patrizio does suggest that additional tabs can be added to a MOF file, Patrizio never teaches or suggests directing such additional tabs to "display names." Indeed, once again, the word "display" never even appears in either of the cited paragraphs.

Appellants next limitation, "creating one or more data elements using the data definitions," is also not taught or suggested by Patrizio. First, the "data definitions" claimed by Appellants correspond to a retrieved "display name." Appellants have already established that Patrizio teaches or suggests nothing regarding receiving or using a display name. Therefore, Patrizio cannot teach "creating ... data elements" using such data definitions. The Final Office Action cites the same paragraphs of Patrizio (paragraphs 38 and 39). Appellants have already established that neither of these paragraphs teach or suggest anything regarding "display" names, nor do either paragraph even contain the word "display."

Appellants final limitation, "writing the qualifier values and data elements to a display panel," is also not taught or suggested by Patrizio. In particular, the qualifier values and data elements both relate to a retrieved display name and Patrizio does not teach anything regarding retrieval of a display name. The Final Office Action cites the same paragraphs of Patrizio (paragraphs 38 and 39) in support of the contention. While these paragraphs discuss modifying a MOF file, neither of these paragraphs even use the word "display." Consequently, neither paragraph teaches or suggests anything to do with a "display panel," and neither paragraph teaches anything regarding writing qualifier names and data elements to a display panel.

As argued above, Appellants have demonstrated that Patrizio does not teach or suggest any of the limitations found in Appellants' independent claims 1, 8, or 15. Therefore, Appellants respectfully request that final rejection of these claims under 35 U.S.C. § 102(e) be **REVERSED** and that Appellants' claims be allowed. In addition, claims 2-7, 9-14, and 16-21 each depend, directly or indirectly, on one of these independent claims and, therefore, claims 2-7, 9-14, and 16-21 are each allowable for at least the same reasons that the independent claims are allowable.

Claim 22 is an independent claim that includes the same limitations as claim 1 with the additional limitations of:



## PATENT

- identifying one or more menu tab names from the retrieved qualifier names;
- creating a menu tab within the display panel corresponding to each of the menu tab names; and
- writing a tab label for each of the menu tabs, wherein the tab labels correspond to the menu tab names.

As an initial matter, claim 22 is allowable because it includes the same limitations as found in claim 1 and, as explained above, Patrizio does not teach or suggest these limitations. Claim 22 is also allowable because Patrizio does not teach or suggest the additional limitations found in claim 22.

The Final Office Action explains that "claims 22, 23 recite various combination (sic) of the limitations recited in claims 2-7, thus are rejected for the same reason as set forth in the rejection of claims 2-7 combined. The additional limitations to claim 22 are also found in dependent claim 4. In rejecting dependent claim 4, the Examiner states that "For creating new menu tabs, descriptions of the new tab are added to the MOF file (0038, 0043). The identifying of menu tab name and writing tab label are inherently included in the teaching of adding new tabs." Appellants respectfully submit that the Examiner's inherency argument is misplaced. The Examiner seems to be mistaking containment "tabs" related to a property sheet taught by Patrizio with "menu tabs" that are used in a display as claimed by Appellants. The two paragraphs cited by the Examiner are reproduced below to buttress Appellants' argument:

[0038] If a change to the layout of a cluster property sheet is required, the modifications are added to the MOF file. For instance, for an additional tab, further descriptions describing another tab are added to the MOF file and the code in the program would not need to be modified. Thus, the instances of the classes are modified in the MOF file, but the schema maintains generality and need not be modified. In one embodiment, the application is programmed using JAVA.TM. (JAVA is a trademark of Sun Microsystems, Inc.). The JAVA.TM. code that exists would read that definition file and would automatically render a new tab. Traditionally, the way this is done is to hard-code it in source code. Thus, JAVA.TM. code to specifically render all of the components needed for the cluster property sheet would need to be written and reintegrated into the existing code. In the preferred embodiment, the desired changes are entered into a pre-processor which checks the syntax and then generates the modified MOF file.

[0043] With reference again to FIG. 9A, the CMGuiPSTabContainment class 912 has a one-to-one relationship with a CMGuiPSTab class 914. This class specifies

## PATENT

that each tab has a label, associated help, and visibility flag. The main purpose of this class is to connect property sheets and tabs. An advantage of this method is that a particular style of tabs could be shared among different property sheets. Moreover, because the help string is associated with a tab, context sensitive help is available at the tab level rather than at just the property sheet level. The visibility flag allows a tab to be made invisible, if desired. This allows a set of users to be blind to some data for security, aesthetic or other reasons. Or more accurately, for a vendor or developer to control which of the tabs are seen by various customers. A tabVisible flag can easily be inverted from false to true to enable a particular customer to see the tab, without having to change their source code.

The containment tabs described by Patrizio are taught as being connected to "property sheets" using a particular class ("The main purpose of this class is to connect property sheets and tabs."). In Appellants' claim 22, the menu tab names are identified based upon the retrieved "qualifier names" that, in turn, were retrieved based upon a "display name" corresponding to an element sought by a user. In Patrizio, the "containment tabs" are "hard coded" into the MOF file (paragraph 39: "Traditionally, the way this is done is to hard-code it in source code. Thus, JAVA.TM. code to specifically render all of the components needed for the cluster property sheet would need to be written and reintegrated into the existing code. In the preferred embodiment, the desired changes are entered into a pre-processor which checks the syntax and then generates the modified MOF file."). Appellants' next limitation, "creating a menu tab within the display panel corresponding to each of the menu tab names," creates the displayed menu tab "on-the-fly" based upon the display name retrieved from the MOF file. In other words, Appellants' claim "creating" the menu tabs in response to data received from a user, while Patrizio teaches that the data in the MOF file is constant and does not teach or suggest receiving an element request from a user, retrieving a display name corresponding to the request, retrieving qualifier values and data definitions corresponding to the display name, identifying "menu tab names from the retrieved qualifier names," and "creating a menu tab within [a] display panel," as taught and claimed by Appellants.

Consequently, as explained above, Patrizio does not anticipate Appellants' claim 22 as alleged in the Final Office Action. Accordingly, Appellants respectfully request that the Board **REVERSE** the final rejection of claim 22.

## PATENT

Claim 23 includes the same limitations as found in claim 1 with the following additional limitations:

- retrieving one or more text labels from the management definition object, wherein each of the text labels corresponds to one of the data elements; and
- writing the text labels in the display panel in a position adjacent to each text label's corresponding data element.

These additional limitations are the same as those found in dependent claim 6. The Examiner rejected claim 6 by stating "New property sheet can be added. The MOF comprises information describing the property sheet, which includes text labels correspond (sic) to the property sheet." It is apparent from the rejection that the Examiner is misconstruing Appellants' "display panels" with "property sheets" found in a MOF file. The cited sections of Patrizio are as follows:

[0039] Referring now to FIGS. 9A and 9B, there is shown a class schema, generally designated by the reference numeral 900a and 900B, for the property sheets, pursuant to the principles and teachings of the present invention. Referring specifically to FIG. 9A, class CMGuiPSheet 910 defines the general property sheet, and has a class identifier (mapClassId), a title string, a title property name string, a version and a default height and width, as illustrated in the FIG. 9A. In the exemplary embodiment, there are three objects having property sheets: cluster, node and package. Therefore, there will be three instances of the CMGuiPSheet class in the MOF file that holds the instances of the defined classes. If a new object is defined, the schema 900a requires no modification. The instance MOF file would be modified to add the new property sheet instance and associated class instantiations.

[0040] The CMGuiPSheet class has a one-to-many relationship with a CMGuiPSTabContainment class 912, which defines a tab for containment within the property sheet. A sheet may have multiple tabs. The sequence of tabs is also defined here. The sequence defines the order in which the tabs appear (top to bottom). The actual sequence is identified in the instance MOF file. Because there is a one-to-many relationship, a property sheet defined to have four tabs will have four instances of the containment class. For instance, in an exemplary embodiment, there are several tab containment instances for a cluster property sheet. The MOF file would therefore include the following instantiations:

```
1 instance of CMGuiPSheet { id = "CMGuiPSheet:SGCluster"; mapClassID =
"SGCluster"; title = "PSMOF_SGCluster_title"; titlePropertyName = "name";
version = "010502" width = 480; height = 420; }; instance of
GMGuiPSTabContainment { id =
```

## PATENT

```

"CMGuiPSTabContainment:CMGuiPSheet:SGCluster+CMGuiPSTab:CMGuiPS
h eet:SGCluster:1"; CMGuiPSheetID "CMGuiPSheet:SGCluster";
CMGuiPSTabID = "CMGuiPSTab:CMGuiPSheet:SGCluster:1"; sequence = 1;
defaultTop = true; }; instance of CMGuiPSTabContainment { id =
"CMGuiPSTabContainment:CMGuiP-
Sheet:SGCluster+CMGuiPSTab:CMGuiPSheet:SGCluster:2"; CMGuiPSheetID
= "CMGuiPSheet:SGCluster"; CMGuiPSTabID =
"CMGuiPSTab:CMGuiPSheet:SGCluster:2"; sequence = 2; defaultTop = false; };
instance of CMGuiPSTabContainment { id =
"CMGuiPSTabContainment:CMGuiPSheet:SGCluster+CMGuiPST-
ab:CMGuiPSheet:SGCluster:3"; CMGuiPSheetID = "CMGuiPSheet:SGCluster";
CMGuiPSTabID = "CMGuiPSTab:CMGuiPSheet:SGCluster:3"; sequence = 3;
defaultTop = false; }; instance of CMGuiPSTabContainment { id =
"CMGuiPSTabContainment:CMGuiPSheet:SGCluster+CMGuiPSTab:CMGuiPS
h eet:SGCluster:4"; CMGuiPSheetID = "CMGuiPSheet:SGCluster";
CMGuiPSTabID = "CMGuiPSTab:CMGuiPSheet:SGCluster:4"; sequence = 4;
defaultTop = false; };

```

As explained above, Patrizio does not teach or suggest the limitations found in Appellants' claim 1. Claims 23 includes these same limitations and, therefore, is allowable for the same reasons that claim 1 is allowable.

As the Examiner states, the MOF file includes information about "property sheets." Appellants' claimed invention, on the other hand, extracts information from specially prepared MOF files in order to retrieve, on-the-fly, information that is written to display panels and displayed to a user. Claims 4 and 23 include the retrieval of "text labels" from the MOF and the writing of such labels to a display panel. However, first the user entered an element and the element was used to find a display panel and then retrieve qualifier values and data definitions corresponding to the display name. As described in the traversal of the rejection of claim 1, Patrizio simply does not teach these limitations.

Consequently, as explained above, Patrizio does not anticipate Appellants' claim 23 as alleged in the Final Office Action. Accordingly, Appellants respectfully request that the Board REVERSE the final rejection of claim 23.

Claims 24 and 25 are both independent claims. Claim 24 includes the same limitations as found in claim 8 and claim 25 includes the same limitations as found in claim 15. Consequently, both of these claims are allowable for at least the same reasons that claims 8 and 15 are

## PATENT

allowable, as discussed above. In addition, claim 24 includes additional limitations found in claims 11 and 13 which are the same limitations as the additional limitations added to claims 22 and 23. Therefore, claim 24 is also allowable for the same additional reasons that claims 22 and 23 are allowable, as discussed above. Likewise, claim 25 includes limitations found in claims 15, 18, and 20 with the additional limitations found in claims 18 and 20 corresponding to the additional limitations found in claims 22 and 23. Therefore, claim 25 is also allowable for the same additional reasons that claims 22 and 23 are allowable, as discussed above.

**The Patrizio Reference is NOT Prior Art to Appellants' Claimed Invention**

The Examiner refused to consider the declaration of Mr. Kevin Barker under 37 CFR 1.131 swearing behind the Patrizio reference. Appellants respectfully submit that the refusal to consider Appellants' declaration is improper.

First, Mr. Barker's declared that he is an inventor of the claims under appeal, namely the independent claims. Requiring all of the inventors to sign the same declaration is not a practice consistently followed by the Patent Office and, in many cases, is onerous and unnecessary. Appellants submit that if the question was posed in a court of law, evidence would properly be considered if offered by one of the inventors so long as the inventor testifying had requisite knowledge of the material to which the inventor was testifying. Rules of practice before a trial court would certainly not require that all inventors need to testify together, as the Examiner states that MPEP 715.04B requires. Moreover, the undersigned attorney has proffered numerous declarations on behalf of the Real Party in Interest. To the undersigned attorney's knowledge, this marks the first occasion that a properly executed declaration under Rule 1.131 was not accepted simply because "all" of the inventors did not sign the same piece of paper. Certainly, one of the Appellants, such as Mr. Barker, is sufficient to simply testify to the attached Exhibits. Indeed, Mr. Barker's name appears on the Exhibits as one of the inventors. Indeed, in this instance, Mr. Barker was the "lead" inventor for the application and, as such, was perhaps the only inventor of the Application that was an inventor of each and every claim rejected as being anticipated by Patrizio. Therefore, the undersigned attorney worked with Mr. Barker on the preparation of Mr. Barker's declaration and accompanying Exhibit. Appellants therefore submit that the Declaration of Mr. Barker as being "the inventor" of the claims rejected as anticipated by

## PATENT

Patrizio was proper under MPEP § 715.04B. Indeed, a review of the actual Rule 1.131 does not require that "all" inventors must sign the declaration. Instead, the Rule simply states that "the inventor" may submit an oath or declaration. Appellants respectfully submit that Mr. Barker qualifies as "the inventor" qualified to sign such declaration.

Regarding Appellants' Declaration, the Examiner states: "While conception is the mental part of the inventive act, it must be capable of proof, such as by demonstrative evidence or by a complete disclosure to another. Conception is more than a vague idea of how to solve a problem." Appellants are truly baffled by the Examiner's statement. Appellants' declaration included a 35 page Exhibit complete with detail regarding Appellants' invention.

Next, the Final Office Action contends that "the evidence fails to show that the inventor constructed an embodiment or performed a process that met all limitations of the claims. Proof of constructive reduction to practice requires sufficient disclosure under how to make and use of 35 USC 112-1<sup>st</sup> paragraph. Appellants respectfully submit that Appellants' 35 page Exhibit to Mr. Barker's declaration is more than sufficient to qualify as a constructive reduction to practice. The Exhibits describe in sufficient detail algorithms used by the software developed by the inventors as well as an "internal project completion" date for the software (see page 4 of patent disclosure). The Exhibit also includes several "screen shots" showing Appellants software being executed on a computer system.

Finally, the Final Office Action state that a showing of diligence must be made. Appellants respectfully submit that the declaration of Mr. Barker states that the Appellants were diligent in reducing the invention to practice. Moreover, the screen shots included in the Exhibit evidence that, to some extent, the inventors already had a constructive reduction to practice before the date of the Patrizio reference. If the Examiner would have requested additional information in a non-final Office Action, Appellants would have been more than willing to provide such information. However, the Examiner elected to finally reject Appellants' claims and Appellants decided to appeal the rejection of their claims based on the fact that the art cited by the Examiner did not teach or suggest Appellants' claimed invention and that the Declaration filed under 37 CFR 1.131 was sufficient to remove the Patrizio reference as a prior art reference.

## PATENT


Appellants have included a copy of the Declaration and the Exhibit thereto for the Board's review and consideration.

In light of the foregoing, Appellants respectfully request that the Examiner's rejection of Appellants' Declaration under 37 CFR 1.131 be **REVERSED**. As the Patrizio reference is not prior art to Appellants' claimed invention, the rejection of claims 1-25 as being anticipated by Patrizio must also be **REVERSED**.

**Conclusion**

For the foregoing reasons, Appellants submit that claims 1 through 25 are patentable over the cited prior art. Accordingly, Appellants respectfully requests that the Examiner's claim rejections be reversed and claims 1 through 25 be allowed.

Respectfully submitted,

By   
Joseph T. Van Leeuwen  
Attorney for Appellants  
Registration No. 44,383  
Telephone: (512) 301-6738  
Facsimile: (512) 301-6742

## PATENT

## APPENDIX OF CLAIMS

1. A method of generating display information from management definition data, said method comprising:  
receiving an element request from a user;  
locating a display name corresponding to the element request in a management definition object;  
retrieving one or more qualifier values and one or more data definitions corresponding to the display name, wherein the retrieving includes reading the management definition object;  
creating one or more data elements using the data definitions; and  
writing the qualifier values and data elements to a display panel.
2. The method as described in claim 1 wherein the management definition object includes a common information model managed object format file.
3. The method as described in claim 1 further comprising:  
associating one of the data elements with an external data source.
4. The method as described in claim 1 further comprising:  
identifying one or more menu tab names from the retrieved qualifier names;  
creating a menu tab within the display panel corresponding to each of the menu tab names; and  
writing a tab label for each of the menu tabs, wherein the tab labels correspond to the menu tab names.
5. The method as described in claim 1 wherein at least one of the data elements is selected from the group consisting of a text box control, a list box control, a combo box control, a check box control, and a radio button control.
6. The method as described in claim 1 further comprising:



## PATENT

- retrieving one or more text labels from the management definition object, wherein each of the text labels corresponds to one of the data elements; and  
writing the text labels in the display panel in a position adjacent to each text label's corresponding data element.
7. The method as described in claim 1 wherein the data definitions include one or more data specifications corresponding to at least one of the data elements, and wherein at least one of the data specifications are selected from the group consisting of a minimum value, a maximum value, a data type, and a valid values list.
8. An information handling system comprising:  
one or more processors;  
a memory accessible by the processors;  
a nonvolatile storage area accessible by the processors; and  
a display generation tool for generating display information from management information, the display generation tool including:  
input logic for receiving an element request from a user;  
identification logic for locating a display name corresponding to the element request in a management definition object;  
means for retrieving one or more qualifier values and one or more data definitions corresponding to the display name, wherein the retrieving includes means for reading the management definition object;  
creation logic for creating one or more data elements using the data definitions;  
output logic for writing the qualifier values and data elements to a display panel;  
and  
storage logic for storing the display panel in the nonvolatile storage area.
9. The information handling system as described in claim 8 wherein the management definition object includes a common information model managed object format file.
10. The information handling system as described in claim 8 further comprising:  
association logic for associating one of the data elements with an external data source.

## PATENT

11. The information handling system as described in claim 8 further comprising:  
identification logic for identifying one or more menu tab names from the retrieved  
qualifier names;  
creation logic for creating a menu tab within the display panel corresponding to each of  
the menu tab names; and  
output logic for writing a tab label for each of the menu tabs, wherein the tab labels  
correspond to the menu tab names.
12. The information handling system as described in claim 8 wherein at least one of the data  
elements is selected from the group consisting of a text box control, a list box control, a  
combo box control, a check box control, and a radio button control.
13. The information handling system as described in claim 8 further comprising:  
retrieval logic for retrieving one or more text labels from the management definition  
object, wherein each of the text labels corresponds to one of the data elements;  
and  
output logic for writing the text labels in the display panel in a position adjacent to each  
text label's corresponding data element.
14. The information handling system as described in claim 8 wherein the data definitions  
include one or more data specifications corresponding to at least one of the data elements,  
and wherein at least one of the data specifications are selected from the group consisting  
of a minimum value, a maximum value, a data type, and a valid values list.
15. A computer program product stored on a computer operable medium for generating  
display information from management definition data, said computer program product  
comprising:  
means for receiving an element request from a user;  
means for locating a display name corresponding to the element request in a management  
definition object;

## PATENT

means for retrieving one or more qualifier values and one or more data definitions corresponding to the display name, wherein the retrieving includes reading the management definition object;

means for creating one or more data elements using the data definitions; and

means for writing the qualifier values and data elements to a display panel.

16. The computer program product as described in claim 15 wherein the management definition object includes a common information model managed object format file.
17. The computer program product as described in claim 15 further comprising:  
means for associating one of the data elements with an external data source.
18. The computer program product as described in claim 15 further comprising:  
means for identifying one or more menu tab names from the retrieved qualifier names;  
means for creating a menu tab within the display panel corresponding to each of the menu tab names; and  
means for writing a tab label for each of the menu tabs, wherein the tab labels correspond to the menu tab names.
19. The computer program product as described in claim 15 wherein at least one of the data elements is selected from the group consisting of a text box control, a list box control, a combo box control, a check box control, and a radio button control.
20. The computer program product as described in claim 15 further comprising:  
means for retrieving one or more text labels from the management definition object,  
wherein each of the text labels corresponds to one of the data elements; and  
means for writing the text labels in the display panel in a position adjacent to each text label's corresponding data element.
21. The computer program product as described in claim 15 wherein the data definitions include one or more data specifications corresponding to at least one of the data elements, and wherein at least one of the data specifications are selected from the group consisting of a minimum value, a maximum value, a data type, and a valid values list.

## PATENT

22. A method of generating display information from management definition data, said method comprising:  
receiving an element request from a user;  
locating a display name corresponding to the element request in a management definition object, wherein the management definition object includes a common information model managed object format file;  
retrieving one or more qualifier values and one or more data definitions corresponding to the display name, wherein the retrieving includes reading the management definition object;  
creating one or more data elements using the data definitions;  
writing the qualifier values and data elements to a display panel;  
identifying one or more menu tab names from the retrieved qualifier names;  
creating a menu tab within the display panel corresponding to each of the menu tab names; and  
writing a tab label for each of the menu tabs, wherein the tab labels correspond to the menu tab names.
23. A method of generating display information from management definition data, said method comprising:  
receiving an element request from a user;  
locating a display name corresponding to the element request in a management definition object, wherein the management definition object includes a common information model managed object format file;  
retrieving one or more qualifier values and one or more data definitions corresponding to the display name, wherein the retrieving includes reading the management definition object;  
creating one or more data elements using the data definitions;  
writing the qualifier values and data control objects to a display panel;  
retrieving one or more text labels from the management definition object, wherein each of the text labels corresponds to one of the data elements; and

## PATENT

writing the text labels in the display panel in a position adjacent to each text label's corresponding data element.

24. An information handling system comprising:
- one or more processors;
  - a memory accessible by the processors;
  - a nonvolatile storage area accessible by the processors; and
  - a display generation tool for generating display information from management information, the display generation tool including:
    - input logic for receiving an element request from a user;
    - identification logic for locating a display name corresponding to the element request in a management definition object;
    - means for retrieving one or more qualifier values and one or more data definitions corresponding to the display name, wherein the retrieving includes means for reading the management definition object;
    - creation logic for creating one or more data elements using the data definitions;
    - output logic for writing the qualifier values and data elements to a display panel;
    - storage logic for storing the display panel in the nonvolatile storage area
    - identification logic for identifying one or more menu tab names from the retrieved qualifier names;
    - creation logic for creating a menu tab within the data panel corresponding to each of the menu tab names;
    - output logic for writing a tab label for each of the menu tabs, wherein the tab labels correspond to the menu tab names;
    - retrieval logic for retrieving one or more text labels from the management definition object, wherein each of the text labels corresponds to one of the data elements; and
    - output logic for writing the text labels in the display panel in a position adjacent to each text label's corresponding data element.

## PATENT

25. A computer program product stored on a computer operable medium for generating display information from management definition data, said computer program product comprising:
- means for receiving an element request from a user;
  - means for locating a display name corresponding to the element request in a management definition object;
  - means for retrieving one or more qualifier values and one or more data definitions corresponding to the display name, wherein the retrieving includes reading the management definition object;
  - means for creating one or more data elements using the data definitions;
  - means for writing the qualifier values and data elements to a display panel;
  - means for identifying one or more menu tab names from the retrieved qualifier names;
  - means for creating a menu tab within the display panel corresponding to each of the menu tab names;
  - means for writing a tab label for each of the menu tabs, wherein the tab labels correspond to the menu tab names;
  - means for retrieving one or more text labels from the management definition object, wherein each of the text labels corresponds to one of the data elements; and
  - means for writing the text labels in the display panel in a position adjacent to each text label's corresponding data element.

**PATENT**

**EVIDENCE APPENDIX**

Declaration of Mr. Kevin Barker and Exhibit thereto has been attached to this Appeal Brief.

Docket No. RSW920010049US1

Page 23 of 24  
**Barker et al. - 10/046,940**

Atty Ref. No. IBM-R109

**PATENT**

**RELATED PROCEEDINGS APPENDIX**

Not applicable.

Docket No. RSW920010049US1

Page 24 of 24  
Barker et al. - 10/046,940

Atty Ref. No. IBM-R109